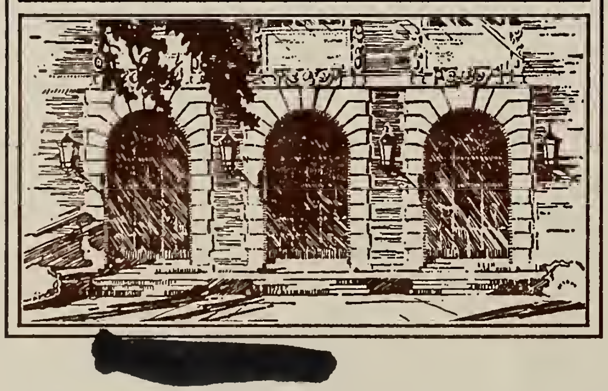


[redacted]

REPORT NO. 467

Math

COO-1469-0189

GRASS: Terminal User's Guide

by

M. J. Michel
J. Koch

August 1971

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

REPORT NO. 467

GRASS: Terminal User's Guide*

by

M. J. Michel
J. Koch

August 1971

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

* Supported in part by the Atomic Energy Commission under Grant US AEC AT(11-1)1469.

TABLE OF CONTENTS

PREFACE

This report is intended as a reference aid describing the operation of a user's terminal on the Graphical Remote Access Support System (GRASS). Familiarity with an overview of the system [1] and with the physical operation of the keyboard and joystick on a Computek graphics terminal [2] is assumed. Further information regarding the system can be found in the remaining references. A sample terminal session is described in Appendix A. Instructions for loading the entire system are given in Appendix B.

The help (and fortitude) of Janet Michel, who was asked to operate the system according to the directions in this report, is hereby gratefully acknowledged.

LIST OF FIGURES

1. TERMINAL USE

Each terminal consists of a direct-view storage tube display, an alphanumeric keyboard, and a joystick cursor control. The system supports concurrent operation of up to eight such terminals.

1.1 Basics

GRASS terminals will accept two types of input: joystick "hits" and text lines. A "hit" causes the X and Y coordinates of the current position of the joystick cursor to be sent to the computer. Pressing any of the six keyboard function buttons or the three interrupt buttons on the joystick box results in a "hit". In all following descriptions, to "hit" means to position the cursor over the item or area of the display desired and then to press one of the previously mentioned buttons. Text lines are constructed character by character in a local terminal line buffer. Pressing RETURN causes the current line to be sent to the computer. Pressing any keyboard button except RETURN or the joystick hit buttons results in the corresponding character being added to the righthand end of the current line. When any characters are in the line buffer, the joystick cursor is disabled. Instead, an imaginary text character cursor is maintained at the righthand end of the textline. Pressing control-BAC moves the text cursor one character to the left (deletes rightmost character); pressing control-K moves the text cursor to the extreme left margin (this "kills" the line in the buffer, no data is sent to the computer, and the joystick cursor is re-enabled). Transmitting the text line with RETURN re-enables the joystick cursor; the text cursor is moved to the extreme left margin. In all following descriptions, to "type" means to construct a text line in the buffer and to transmit the line to the computer by pressing RETURN.

The segmentation of the display area imposed by GRASS is shown in Figure 1. In various modes, certain of these areas are disabled. That is, the system expects a hit only in some particular sections; hits in other areas will be completely ignored. Whenever the system regenerates

the display (e.g., issues an erase followed by a new display), the time of day is shown in the upper righthand corner of the screen. Messages giving the terminal user directions, the status of the system, or the name of the current mode generally appear in the "prompt" and "system message" areas. Lists of mode options are generally presented in the "menu" area, while functions to be chosen are placed in the "light button" area. The "Return" area usually causes exiting from a current mode to the calling mode. Certain additional responses or options are shown in the "Answer" area. The "panic" area is used to re-initialize a mode or function after a mistake (e.g., "hitting the panic button").

1.2 Log-On/Log-Off

The system will in general be available for operation; only the activation of individual terminals is ordinarily necessary. If the system must be loaded from scratch, consult Appendix B.

Before attempting to use the system, the terminal must be made ready. If the ØN button is not lit, press it once. Make certain that the LINE button is now lit; it if is not, press it once. After waiting thirty seconds for warmup, press ERASE, then RESET. Ask the machine operator to "enable" the terminal (or, turn the switch on the PDP-8/I teletype to LINE, type "nE", and turn the switch off. The "n" is the terminal number (0-7). If the terminal is powered off (by pressing ØN once again), type "nD" on the teletype to disable the terminal). When the operator enables the terminal, the screen will flash (erase) and the READY light will come on. If the ØN button had been originally hit (and READY was lit) all of the above steps would be omitted. Operation can now begin.

Press RETURN or cause a hit: the system frame will appear with the message 'TYPE "PSNØNAMEABCD" (LIMIT=2 MIN)'. Type a valid logon identifier. This consists of a four digit number (with leading zeroes) followed by a character string. An example is: "0101JØNES." While logging-on, hits are ignored for all screen areas. If the identifier is mistyped or not on file, the message "NØT FOUND: PLEASE TRY AGAIN" will appear. The incorrect identifier along with the terminal number and time

of day is printed on the system log ("threat" monitoring). No activity at the terminal for two minutes will result in the screen being erased; the system then assumes that the terminal is not in use.

The message "PICK PRØG ØR LØGØUT" will appear after an identifier is successfully recognized. The terminal number, time of day, and scrambled identifier are listed on the system log along with a code indicating "log-on." A list of the programs currently available in the system is placed in the menu area, and "LØGØUT" appears in the panic area. To begin execution of any program, simply hit the desired name. When the chosen program is finished executing, this mode will be re-entered.

To terminate terminal operation, hit LØGØUT. The terminal number, time of day, and scrambled identifier are listed on the system log along with a code indicating "log-off." No activity at the terminal for five minutes while this mode also results in LØGØUT. An alternate method of logging-off is activated by typing "PURGE." In this case, a standard LØGØUT is performed, but all files saved locally during the session are deleted from the disk. This leaves more space available for other users still on the system.

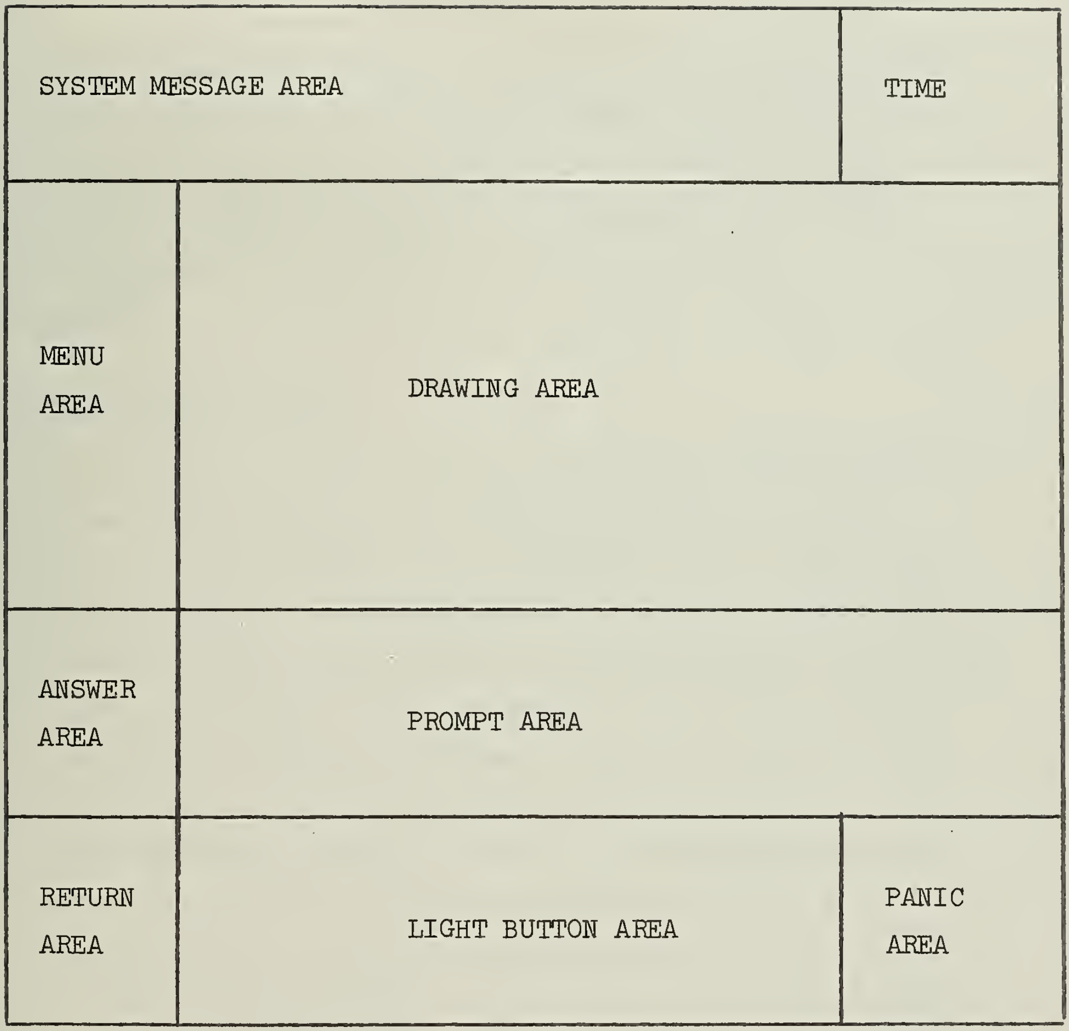


Figure 1. Display Area Organization

2. SYSTEM SUPPLIED PROGRAMS

Two highly generalized support programs are provided by the system. Due to the modularity of the system, these programs cannot only invoke each other, but can also be called as subroutines from user-written applications programs. Moreover, most individual functions within these two programs can be called as subroutines from applications programs.

2.1 Generalized Drawing (GNDRW)

This program provides the ability to create pictures composed of arbitrary lines, text segments, and mnemonic instances of other pictures. Mnemonic instances may include terminals (e.g., connection points) and a parameter list. Terminals may be connected together in any arbitrary manner; hence, relationships between the mnemonic instances of a picture can be established. The parameter list for each mnemonic instance can contain any arbitrary text data; hence, information (such as variable assignments) can be "passed" to the picture that is represented by the mnemonic instance. Note that a picture may contain all, none, or any combination of the above as desired by the user.

When GNDRW begins execution, the system frame appears with the system message "PICTURE CØNSTRUCTIØN." This initial mode is the main control mode for GNDRW; most functions are performed, and all functions are requested, while in this single mode. The MENU area will contain a list of the user's available pictures and corresponding mnemonics if any have been previously saved on the local disk by GNDRW. A hit on RETURN causes execution of GNDRW to terminate; a hit on PANIC resets GNDRW to its initial state. PANIC resets any flags that might be set and then performs a regeneration of the display. Special instructions for the user will appear as necessary in the PRØMPT area. The ANSWER and LIGHT BUTTØN areas contain functions to be chosen. Additional functions can be invoked via typed commands.

The picture being constructed or modified (e.g., the "current" picture) is displayed in the DRAW area. Since the program pick mode of the log-on monitor always initializes all control and data blocks for the terminal before starting the chosen program, the DRAW area will be blank when GENDRW is started from log-on. Note that returning to the log-on monitor will cause the current picture to be lost. Of course, GNDRW can be called as a subroutine from user applications programs to create or load a picture and then called repeatedly to modify that picture (c.f. GSAMV1 in 3.1).

The line drawing function is always in effect when no other function is current. Requesting any new function automatically cancels whatever function is current. Completion of a function returns GNDRW to its initial (e.g., "line drawing") state. Note that some functions are "completed" only when another function is requested. Consult Figure 2 during the following discussions.

RE∅RG, RE∅RG B, RE∅RG E, and Line Drawing

Many GNDRW functions, such as line drawing, refer to the "current cursor position"; this is defined as the last recognized hit in the DRAW area. For example, after a request for adding a mnemonic instance to the picture is made, the mnemonic will be displayed with its center at the current cursor position. Although the current cursor position (hereafter referred to simply as "cursor") is moved implicitly by many of the functions described below, three explicit functions for moving the cursor are provided. These are "RE∅RG," "RE∅RG B," and "RE∅RG E" and are located in the ANSWER area. A hit on RE∅RG ("re-origin") at any time means that the next hit--if in the DRAW area--will position the cursor at that hit. If the next hit is not in the draw area, the RE∅RG request is ignored. During line drawing, a hit on RE∅RG B ("re-origin back") resets the cursor from the end of the line just drawn back to that line's beginning. Repeated hits on RE∅RG B are ignored. Hitting RE∅RG E ("re-origin end") at any time after line drawing resets the cursor to the end of the last line drawn. Hence, some

lines can be drawn, then some mnemonics and text can be added, and then line drawing can be restarted exactly at the point where discontinued. Remember that REØRG, REØRG B, and REØRG E are indeed true functions: after using them, GNDRW is in the drawing mode. Note in addition, that on initial entry to GNDRW or after a PANIC hit, GNDRW responds as if REØRG had just been hit. That is, the next hit will reset the cursor if the hit occurs in the DRAW area.

When in the line drawing mode, a hit in the DRAW area causes a line to be placed beginning at the cursor and ending at the hit. The cursor is then moved to the position of the hit (implicitly moved). A subsequent DRAW area hit causes a second line to be placed beginning at the cursor (e.g., the end of the previous line) and ending at the hit; the cursor is again moved to the hit. To start a new line group that does not continue from the cursor, use REØRG to set the initial end-point, then proceed as above.

Typed Commands

All functions that save, fetch, or delete pictures and mnemonics on the local disk can be invoked only from typed commands except for one function. Mnemoncis can be fetched by typing the appropriate command or by hitting its name in the menu area. In the following discussion, ".x" implies a period followed by any single digit, 0 through 7. A mnemonic name is specified by an eight character string "name.x" where "name" must be padded on the right with blanks if necessary to make it six characters long. A picture name is specified by "name"; no padding of blanks is necessary for picture names.

S#name: Save the current picture on the local disk with the identifier "name." The file control block and any of the ten constituent blocks that contain valid data are written to the disk. Data is only copied; the current picture is not lost. The contents of the menu is updated with the entry "name"; the message "DØNE" appears when the operation is complete. If the menu was already full (128 entries), the picture would be saved but "name" would not be added to the menu.

S&name.X: Save the mnemonic, just created, with the identifier "name.X". Action is the same as for "S#name". Note that the typed command "C&" is used to create a mnemonic; a full discussion of its use is contained in the discussion "Mnemonic Creation" found below.

D#name,D&name.X: Delete a picture or mnemonic from the disk. The appropriate name is also removed from the menu; "DØNE" appears when the operation is complete.

##name: Fetch the designated picture from the disk and make it the current picture. The current picture is destroyed before the designated picture is fetched. If the designated picture does not exist on the disk, then GNDRW is re-initialized with a blank current picture. This is convenient for starting construction of a new picture.

&&name.X: Fetch the designated mnemonic from the disk and add it as an instance to the current picture at the cursor location. As previously mentioned, hitting the name in the menu is the equivalent of this command. The mnemonic will be added to the display, but its terminals will not be visible until regeneration of the picture is subsequently requested. The message "ERRØR" appears if the designated mnemonic does not exist on the disk. Note that "ERRØR" will also appear if the name of a picture was accidentally hit in the menu. Pictures may not be fetched by menu hit since an accidental hit would cause the current picture to be lost.

!": Invoke REACT as a subroutine. Consult the description of the remote communication program in section 2.2. The current picture is not affected unless a remote user program transmits a new picture for saving or modification during REACT processing.

C&: Create a mnemonic from the current picture. Consult the discussion "Mnemonic Creation" found below. The current picture is not affected.

anything else: Enter the text editor to add a comment text segment to the current picture at the cursor position since no command was recognized. Consult the discussion "Text Editor" found below.

Immediate Functions

A hit on any of the functions represented by the string of single characters in the top line of the LIGHT BUTTON area will produce an immediate response. No further input is needed for the completion of these functions. A hit on an illegal or unimplemented function will be ignored. The functions are described in order of appearance, left to right.

<, A, V, >: add the appropriate directional arrow to the current picture at the cursor.

O: add a local terminal to the current picture at the cursor. (c.f. "Mnemonic Creation")

P: display the previous group of 32 names of the menu.

N: display the next group of 32 names of the menu.

B: display a box for aid in creating a mnemonic. The center of the box is the center of the mnemonic; the positions most suitable for terminals on a two-terminal mnemonic are marked; and the size of the box is also appropriate.

X: display all terminal connections in the current picture whether visible or invisible.

L: delete all lines of the current picture produced in line drawing mode.

C: delete all comment text segments of the current picture created by the text editor.

Delayed Functions

A hit on any of the functions represented by the string of single characters in the bottom line of the LIGHT BUTTØN area will result in entering a function requiring further input. This input may be one or more hits in the DRAW or MENU areas or text lines. Some functions are only completed by choosing another function. A hit on an illegal or unimplemented function will be ignored. The functions are described in order of appearance, left to right. Note that for these functions, a short dash appears on the display to indicate acceptance of a hit. This is a feedback mechanism for such things as marking the position of the first of two hits, marking an option, and marking the current function.

V: visibly connect terminals. Each pair of terminals indicated by two subsequent hits is connected with an entry in the data structure and a visible line is displayed between them. A dash will mark the first terminal; the line will appear after the hit on the second terminal. If a hit is not sufficiently close to the center of the terminal symbol to allow identification, no response is made; that is, the expected dash or line will not appear. Another function must be chosen in order to exit this mode, since all draw area hits are assumed to be aimed at identifying terminal pairs.

I: invisibly connect terminals. Identical to "V" except that the connection is not shown by a visible line, and the second terminal in a pair is marked by a dash.

E: mark terminals external. Each successively hit terminal is marked as external. On the next requested display regeneration, the first digit of the numbers on the selected terminals will indicate the change. Another function must be chosen in order to exit from this mode. (c.f. "Mnemonic Creation")

U: mark terminals unexternal. Each successively hit terminal is marked as internal. Action is the same as for "E."

T: assign terminal types. The list of terminal type names available for assignment to local (as opposed to "mnemonic") terminals is displayed in the MENU area. First hit a terminal type name; then all successively hit local terminals will be marked by having a number corresponding to the name entered into the data structure. If terminals are hit before any name in the list has been hit, the selected terminals are marked with a code indicating "undefined type." After hitting one name and selecting a few terminals, the user can hit a different name in the list and select other terminals, thus assigning them the other name. Any number of name and terminal selections can be made. However, the last name assigned to a terminal is the one that is retained. If no names are in the list, names may be added with the following function (amend type name list). A hit in the RETURN area causes termination of the function and return to the main GNDRW mode. Text lines are ignored while in this function.

A: amend terminal type name list. The text editor is entered and the list of names to be modified, if any, is displayed in the MENU area. This list specifies the type names available for assignment to local terminals and is used in the "assign-type-name" function described above. Only one name may be on a line, and once any type assignments are made, no name should be deleted from the list (the assignment numbers would no longer match the intended name). The user may create any names desired; this and related functions merely provide a vehicle to aid the user in delineating different terminal categories

for his own needs. A RETURN hit terminates the function and the main GNDRW mode is re-entered.

L: locate terminal type name. The terminal type name, if any, assigned to each subsequently hit terminal will be displayed. Another function must be chosen to exit from this mode.

D: disappear terminal number. The number on each subsequently hit terminal will not be displayed during future picture regenerations. This is an aid for uncluttering dense pictures. Another function must be chosen to exit this mode.

R: reappear terminal number. The number on each subsequently hit terminal will be displayed during future picture regeneration. This undoes a "D" described above. Another function must be chosen to exit this mode.

M: merge terminals. Merge all subsequently hit terminals into a composite terminal. Only one composite can be made for each call to this routine. Another function must be chosen to exit this mode. Merged terminals will appear as a single terminal, at the position of the first terminal selected, on future picture regenerations. This is a viewing aid.

P: partition terminals. Partition all subsequently hit composite terminals into their constituent terminals. Another function must be chosen to exit this mode. This parameter is a viewing aid.

O: display original mnemonic parameters. The parameter model list for each subsequently hit mnemonic instance is displayed. Another function must be chosen to exit this mode.

A: alter assigned mnemonic parameters. Selecting a mnemonic instance with a hit causes entry to the text editor and display of the parameter list for the chosen instance. This list can be modified with the text editor. RETURN from the text editor causes termination of the function and re-entry to the main GNDRW mode.

A: add comment text. The text editor is entered; text lines typed and modified with the editor will appear as a contiguous comment text segment on the current picture at the cursor position. RETURN from the editor causes termination of the function. As mentioned previously, an unrecognized command text line causes this same function to be invoked.

C: change comment text. A subsequent hit on the upper lefthand character of a comment text segment will cause entry to the text editor. The selected segment can be modified with the editor. RETURN from the editor causes termination of the function.

T: move terminal. Hit pairs redefine the position of local terminals. The first hit specifies the position; the second hit specifies the terminal to be moved to that position. Another function must be chosen to exit this mode.

I: move instance. Hit pairs redefine the positions of mnemonic instances. Action is the same as for "move terminal."

C: move comment text segment. Hit pairs redefine the position of comment text segments. Action is the same as for "move terminal."

L: move line end. Hit pairs redefine the position of line end points. Action is the same as for "move terminal." Note that <u>all</u> line ends in the "vicinity" of the second hit will be moved and that <u>all</u> will then have exactly the same end point. This is very useful for moving a vertex and for "closing up" vertex lines that do not quite match as intended.

T: delete local terminal. Each subsequent local terminal hit is deleted as well as all connections to it. Another function must be chosen to exit this work.

I: delete instance. Each subsequent instance hit is deleted, along with its terminals and all their connections. Another function must be chosen to exit this mode.

C: delete comment text segment. Each subsequent text segment selected by a hit on its upper lefthand character is deleted. Another function must be chosen to exit this mode.

L: delete lines. Hit pairs defined the end points of the line to be deleted. However, if the first hit adequately identifies the line, the deletion will occur after only one hit. Another function must be chosen to exit this mode.

X: delete connection. Hit pairs on terminals define the connection to be deleted. However, if the first hit adequately identifies the connection, the deletion will occur after only one hit. Another function must be chosen to exit this mode.

Text Editor

The GNDRW text editor, a very heavily used subfunction, is capable of modifying any block of contiguous text lines. It is always entered with a parameter pointing to the block to be modified. The default mode for the text editor is "Append"; as each line is typed by the user, it is added at the bottom of the block. Delete, Change, and Insert modes are also available. Two display functions are used with the editor, REGEN and REGEN TXT. The first causes a regeneration of the entire display, including the text being modified and the current picture. The second regenerates only the text being modified, a highly useful function when the picture is quite complex.

Delete, Change, and Insert require at least one prior hit on a line in the text block. After the requisite hit, a hit on the function starts its execution.

DELETE: delete the line hit. If a pair of hits is made before DELETE is chosen all lines between (and including) those hit will be deleted. The second hit of the pair must be on a line below the first. The picture is regenerated, and the editor returns to the APPEND mode.

INSERT: insert all lines subsequently typed above the line hit. This mode is only terminated by a hit on the text block or the APPEND option. Note that the display is not updated after each line typed, but only when REGEN or REGEN TXT is chosen.

CHANGE: delete the line hit from the block, but copy it into the text line buffer at the terminal; then enter INSERT mode. Hence, the line can be modified, and when sent by a RETURN character, will be replaced in the block. Lines typed subsequently will be inserted after the first line returned. If a second hit is made in the block on any line below the first hit before CHANGE is chosen, all included lines are deleted. The first line is still the one copied to the terminal line buffer. Hence, many lines can be replaced by a few, or few by many. Exit from this mode is the same as from INSERT. As with insert, the new lines are only displayed when REGEN or REGEN TXT is chosen.

APPEND: add lines to the end of the block. Text line display is the same as for INSERT.

A hit on RETURN causes termination of the text editor; the calling program is re-entered.

Mnemonic Creation

A picture can only appear as an instance in another picture in the form of a mnemonic. That is, a small simplified drawing that represents the picture is created; whenever the picture is to be used as an instance, its representative mnemonic appears instead. The mnemonic is created from the lines, comment text, external terminals, and terminal type name list of the picture it represents. When an instance is desired, the mnemonic is integrated into the receiving picture.

The mnemonic for a picture can be formed in a fairly general manner. This allows different orientations of the representation to be made. For example, there is a picture that is the definition of a transistor. However, there can be several mnemonics for the picture, one vertical, one horizontal, one reflected in X, one reflected in Y, and so on. The picture and each mnemonic have the same six character name, but the trailing digit on each mnemonic specifies its orientation. Hence, the desired orientation is easily chosen by the user.

Terminals must be discussed at this point. Only certain connection points on a picture are really intended for access to the "outside world." When the picture is used as an instance in another picture, connections to the instance must be made at anticipated or legal positions. For example, connection to a filter must be made at its inputs and outputs, not at positions between the filter's constituent resistors and capacitors. Such terminals of a picture are denoted as "external." All other terminals on the picture, whether added singly (local) or added as part of an instance mnemonic, are termed "internal." Currently, terminals added as part of an instance mnemonic may not be marked as external.

Mnemonics are created as follows. A picture, complete with its external terminals, terminal type name list, constituent instance lines, and comment text, is constructed by GNDRW and saved on the disk with the "S#name" command. The current picture now represents merely a copy of the saved picture. Delete the lines and comment text with "L" and "C" functions; display the box with the "B" function. Now using the normal line drawing and comment text functions of GNDRW, form a simple representation of the picture approximately within the area bounded by the box. When finished, use the terminal move ("T") function to place the external terminals in the positions in the box desired for the mnemonic. Next, use REØRG to move the cursor to the desired position in the box where the parameter list for the mnemonic will appear. Finally, type the command "C&." This causes entry into the text editor; type in the parameters that are desired for the parameter model list.

Note that just before calling the text editor to create the parameter model list, GNDRW extracted the lines, comment text, external terminals, and terminal type name list from the picture and created a mnemonic in a system work block. After the RETURN from the text editor, this mnemonic block is complete. Simply type the "S&name.X" command to store the mnemonic just created. To produce another orientation, begin by using "L", "C", and "B" again, and continue in a similar manner. At any future time, an additional mnemonic can be created for a picture simply by loading the picture with the "##name" command and proceeding as described above.

Other Information

At any time during GNDRW processing, the message "REMØTE DATA FLUSHED" or certain other messages may appear at the top of the DRAW area. These are messages sent to the terminal by the remote computer. If necessary, REACT can be called for responding to or investigation of the messages. Consult 2.2.

During heavy use of the system, a nonresolvable resource lockout may occur. In most instances, the message "LØCKØUT/EXPAND" will appear; sometimes, all that occurs is the loss of part of a picture during regeneration (the display function could not continue; no part of the picture itself is lost.) If this happens, just retry the attempted operation that was in progress at the time of the message or loss. Under some unusual circumstances, the program mode must be exited and reentered to reset the individual terminal. If the error message still persists, the current picture on the terminal has a block that is too big for the computer's memory.

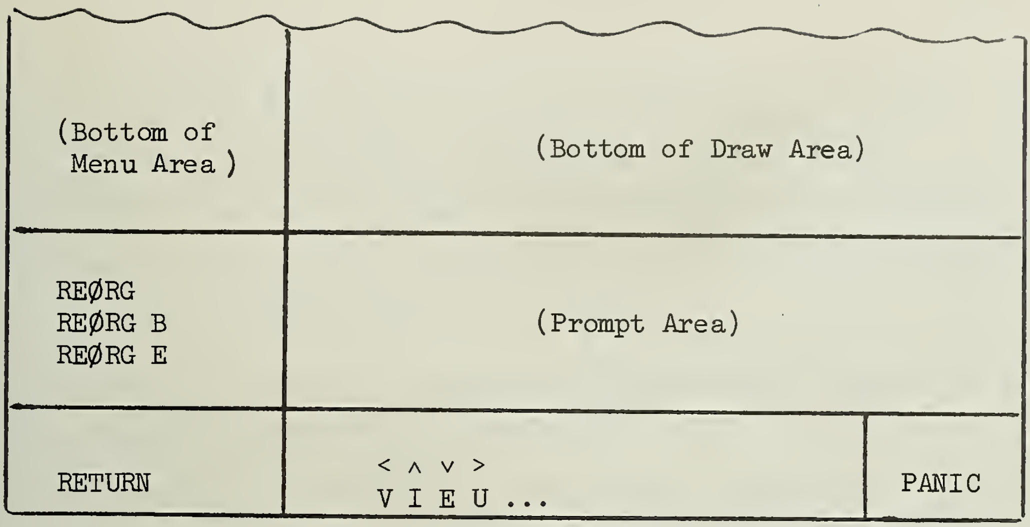


Figure 2. Function Selection Portion of GNDRW Main Mode Display

2.2 Remote Communication (REACT)

This program provides convenient interaction with the large remote computer, in this case, a 360/75. While in this mode, hits and text lines input from the terminal are sent directly to the 360. In addition, certain properly identified text lines cause the PDP-8 to transmit automatically entire pictures or mnemonics that were created and saved by GNDRW. A final, and very powerful, feature of this program is its ability to display and/or automatically save pictures and mnemonics transmitted to the PDP-8 from the 360. No intervention from the terminal user is required; all that is necessary is for programs in the 360 to set the proper control bits in the data transmitted.

When REACT begins execution, the system frame appears with the system message "REMØTE CØMMUNICATIØN." Hitting the RETURN area or the SYSTEM MESSAGE area results in termination of the program. Also, typing "!"" will cause execution of GNDRW to commence. This is a special local command for user convenience; the same result could be obtained by hitting RETURN and then choosing GNDRW for execution. Other local commands are:

##name: load the picture "name" as the current picture; then transmit it to the 360. The picture was previously 1) created and saved by GNDRW or 2) transmitted with the automatic save option from the 360.

&&name: Load the mnemonic "name" as the current mnemonic; then transmit it to the 360. The mnemonic was previously obtained in the same manner as described above for pictures.

C#name: transmit the current picture (already loaded) to the 360, but change its identification to "name."

C&name: transmit the current mnemonic (already loaded) to the 360, but change its identification to "name."

C#: transmit the current picture as above, unchanged.

C&: transmit the current mnemonic as above, unchanged.

All other text lines and all other hits are sent directly to the 360. During normal operation, all lines typed by the user are echoed on the display after being transmitted to the 360. If the echo does not appear immediately, it simply means that the 360 is busy and has not yet accepted the typed line. DØ NØT type another line or cause a hit until the echo from the previous line appears. In addition, the message "DØNE" appears after a picture or mnemonic is successfully transmitted; "ERRØR" appears if transmission is unsuccessful (data error or "name" not found locally).

Of course, before transmitting anything to the 360, it is necessary to have a program executing remotely to accept the transmissions. GRASS has a monitor in the 360 capable of starting a program for each terminal. All data sent to the 360 is examined by the monitor; any typed line beginning with a "!" is considered to be a 360 monitor command. Any data, except monitor commands, sent to the 360 when a program is NOT running under the monitor for the transmitting terminal will automatically be flushed (ignored). Available monitor commands include:

!idstring: Logon request (e.g., "!WSMITH"). The terminal user must successfully identify himself to the remote monitor. All other commands are ignored until this command is accepted. An invalid idstring will be indicated by the message

"INCØRRECT LØGØN"; successful identification is indicated by the message "READY." Other commands may now be entered. Note: since a program is not yet executing for the terminal under the 360 monitor, any data other than monitor commands will still be flushed.

!Sname: Start remote execution of program "name." After the program has begun, the message "TASK STARTED" will appear. Whenever the program completes execution, the message "TASK HALTED XXXXXXXX" will appear; "XXXXXXXX" will be the 360 system completion code for the terminated program. If a program was already executing, the message "ALREADY STARTED" appears.

!DUMP: Print a core dump for the executing program. This is a debugging aid for user convenience. When finished, the message "TASK DUMP CØMPLETED" appears; if no program was executing, the message "NØNE EXECUTING" appears. This command in no way affects the executing program.

!HALT: Request the executing program to terminate. This command causes the monitor to "post" the executing program (c.f. [3]). The message "HALT REQUESTED" will subsequently appear. When (or if) the program accepts the "post" and terminates, the "TASK HALTED" message as described above will appear. If no program was executing, the message "NØNE EXECUTING" appears.

!ZAP: Unconditionally terminate the executing program. This command is used only if a program has not responded to a "!HALT" or if the program is known to be abnormally looping. A "TERMINATIØN REQUESTED" message will appear, followed by a "TASK HALTED" message. If no program was executing, the message "NØNE EXECUTING" appears.

!LØGØUT: Terminate remote processing for this user. This command is only valid when no program is ececuting under the monitor for the terminal. If the command is accepted, "GØØDBYE" will appear. Remote support for the terminal can only be restarted after logout by a new logon command.

!anyother: Illegal command. Any mistyped or unrecognized commands cause "CØMMAND UNKNØWN" to appear.

If the GRASS monitor itself is not executing in the remote computer, no response will be made to the logon command. Since the remote computer will always respond within two minutes, even when very busy, a "no response" is easily distinguished from a "slow response."

The automatic display/store features of REACT are better understood when viewed from the perspective of the 360 programs that utilize them, particularly since no action by the terminal user is needed. This material is presented in [3]; an example of such a program is the Library Service Discographer described in that reference.

3. APPLICATIONS PROGRAMS

The local system currently has provision for online use of approximately twenty user written applications programs. The programs have access to the system programs and to all internal facilities provided by GRASS that are available to the system programs. In addition, interactive programs can execute under the GRASS monitor in the remote 360.

3.1 Simulation and Modeling (GSAMVI)

This program provides for the assignment of parameters, variables, and equations to networks, and to the constituent primitives of networks. Finished networks are transmitted to the remote computer via REACT for mathematical analysis. A description of the analysis program that runs under the GRASS remote monitor, is given in [5].

When GSAMVI begins executing, the system frame appears with the message "SIMULATIØN AND MØDELING (SAM) SYSTEM." Termination of the program can be caused by a hit in the RETURN area, while choosing another mode can be accomplished by hitting one of the three options listed in the MENU area. All other hits and all text lines are ignored at this point. The available modes are "CONSTR", "SPECFY", and "REACT."

The use of the "REACT" option is quite straightforward; choosing this mode simply invokes REACT. Communication of completed networks (e.g., pictures), analysis information, and analysis commands between the remote analysis program and the terminal user is performed as previously discussed in the REACT program description.

Constructing networks in the "CØNSTR" mode is also straightforward; choosing this mode simply invokes GNDRW. For the purposes of the analysis, a picture not containing mnemonic instances of other pictures is a primitive network; a picture that does contain mnemonic instances of other pictures is a network composed of subnetworks. During the first phase of analysis, each subnetwork is examined and evaluated to determine if it is primitive or is composed of subnetworks. The process continues until all subnetworks

are partitioned into their constituent primitives. Connections between terminals of the mnemonic instances in a picture are considered as the nodes between subnetworks in a network. Parameter strings assigned to mnemonic instances in a picture are considered as the values to be passed to subnetworks in a network. The following list of data is all that must be added to a picture created by GNDRW to enable that picture to be analyzed as a network (c.f. [5]).

1. Equations: On primitives, these describe the analytic function of the primitive. On networks, these describe additional constraints on the network.

2. Global and Local Variables and Parameters: These enable the equation parser in the analysis program to symbolically examine the equations. The parameter strings passed to the subnetworks are similarly examined.

3. Terminal Type Declarations: Only certain types of variables may appear on the same terminal. The analysis program must be informed which variables are legal on which terminals.

The "SPECFY" mode provides these services:

Upon entry to the "SPECFY" mode, the system frame appears with the user prompt "CHØØSE AN ØPTIØN ØR RETURN." A list of options is shown in the MENU area. Choosing an option with a hit merely invokes the text editor of GNDRW. Exiting from the text editor causes a return to the "SPECFY" mode, where a different or even the previously chosen option can be requested. Hitting the RETURN area causes re-entry to the main option mode of GSAMVI. While in "SPECFY" (e.g., not in the text editor), all other hits and all text lines are ignored. When all desired information has been added with "SPECFY," the user should return to GSAMVI, choose "CONSTR," and save the picture (e.g., both the part created by GNDRW and the part created by SPECFY is saved). The following information should be entered with the text editor for each "SPECFY" option.

1. DEFS: Terminal type declarations of the form
   Typename: E(E type variable names), I(I type variable names)

   For example: ELECTRIC:E(VØLTAGE),I(CURRENT)
   WATER: E(PRESSURE),I(HEAT,FLØW)

Only the first use of typenames and their respective legal variables need be indicated with DEFS. That is, subsequent pictures using the same typenames and variables should not have these entries in DEFS. Typenames entered into the local typename list of a picture with GNDRW while constructing a primitive (e.g., picture with only local terminals and no mnemonic instances) naturally should correspond to typenames specified by DEFS entries.

2. GLOBAL: Global variables used in equations and parameter strings of the form

   variablename,..., variablename

   For example: TEMP,HUMIDITY,LIGHT

Note that E and I type variables are NOT entered in this list.

3. LØCALS: Local variables used in equations and parameter strings of the form

   variablename,..., variablename

   For example: X, D1, D2, DEL, K

Note that E and I type variables are NOT entered in this list either.

4. SPECLS: Special variables to be processed by the analysis program of the form

   special variablename=identifier

   For example: CLØCK=TIME

which sets the variable CLØCK equal to the system variable TIME. CLØCK must now be processed in a special manner.

5. EQATNS: network or primitive equations of the form

expr = expr,..., expr = expr

For example: HEAT(0)=-HEAT(1)+C,XTEM=R*2*FLØW(3)

All blanks are ignored; maximum length of an equation is 72 characters. Any equation longer than 72 characters must be broken into parts. E and I type variables are identified by the terminal number that follows each such variable in parentheses. For the above example, if R was a global variable, it would be listed in the GLØBLS entry. Similarly, if XTEM was a local variable, it would appear in the LØCALS entry; if C was a parameter, it would appear in the PARMS entry (see below).

6. PARMS: network or primitive parameter strings of the form

$$\left\{\begin{matrix}\text{Parmname}\\ \text{Parmname=expr}\end{matrix}\right\},\ldots,\left\{\begin{matrix}\text{Parmname}\\ \text{Parmname=expr}\end{matrix}\right\}$$

For example: C,VAL,R=100,CØLØR

Note that "R=100" specifies a default value for the parameter "R." If this network is to be used as a subnetwork, this PARMS entry should contain the same data as will be placed in the parameter model list of the mnemonic for this network (c.f. GNDRW). Whenever the mnemonic is used as an instance in a network, the proper parameter list will be available for the later analysis phases. For example, let the parameter list in one instance of the mnemonic be changed to

C=10,VAL=5.

Then, during analysis, this instance of the network will be passed the values

C=10,VAL=5,R=100,CØLØR

Note that CØLØR is undefined since no default was assigned in the network (e.g., in the PARMS entry) and no value was assigned in the instance. Also, R=100 because no different value was assigned in the instance, so the network default was used.

The simulation and modeling application is supported within GRASS by the above very simple program. Only some primitive input/output handling is necessary plus calls to the system programs and to system provided functions (text editor, etc.). Other applications can be supported in a similar manner.

3.2 Other Programs

Several new applications programs are being designed. As yet unrecognized applications could be implemented following suggestions from users.

A text editor with sophisticated commands and file handling ability is being specified. It would be used for document preparation and formatting.

Design and instructional aids for architecture and engineering student use are under consideration. Online picture capabilities obviously indicate applications similar to this.

Graphical page layout for newspapers, magazines, and other publications is being studied.

And, of course, interactive applications programs executing in the 360 and using the automatic facilities of "REACT" are being studied (c.f. [3]). These include graphical language processors, information retrieval packages, algebraic manipulators, various analysis tools, etc. The analysis part of the simulation and modeling package falls into this category.

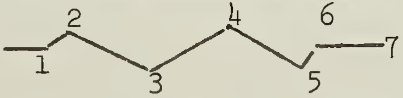
APPENDIX A

Sample Session

This section illustrates the use of the generalized drawing package (GNDRW) to construct and save a simple resistor and two mnemonics. The use of the simulation and modeling program (GSAMV1) is then used to assign information to the resistor picture for use by the corresponding analysis programs that run on the remote computer.

A. Build a resistor (*=joystick hit)

1. Walk up to the Computek terminal, and press the return key.
2. Logon (i.e., type "0101 JØNES").
3. *GNDRW--select the generalized drawing program from the menu list.
4. *B (in light button area). This will cause a box to appear in the center of the screen to help align the drawing.
5. *REORG, *POINT A. (see Figure 3)--get a new origin.
6. *Ø (in light button area). A terminal will appear at point A.
7. *REORG, *POINT B (see Figure 3).
8. *Ø--second terminal appears at point B.
9. *E, *POINT A, *POINT B--makes both terminals external. There should be a small flash on the screen when a hit is made on A and B.
10. *PANIC--screen is regenerated. Note that the terminal numbers changed from 2000, 2001 to 3000, 3001 indicating that they are external. The box has also disappeared.
11. *REORG, *POINT A--use the line drawing mode to draw the picture for a resistor by making a joystick hit on the end points of the lines composing the desired image, e.g.,



12. *REORG, *POINT C, press RETURN key--in text editor mode. Type "THIS IS A RESISTOR:" followed by the RETURN key.
13. *REGEN TXT--only the text line is regenerated.

14. *REGEN--both the picture and the text line are regenerated.
15. *RETURN--back in GNDRW in line drawing mode.
16. *A (append types in light button area)--in text editor. Type ELECTRIC followed by the RETURN key.
17. *REGEN--ELECTRIC appears in the upper left of the screen.
18. *RETURN--back to GNDRW.
19. *T--assign types.
20. *ELECTRIC (in menu area), *POINT A, *POINT B--the terminal type ELECTRIC is now assigned to both terminals.
21. *RETURN--back to GNDRW.

The picture is finished, but it must be saved on the disk. Type S#name followed by the return key where "name" is any six character string. This will save the picture under "name."

B. Construct mnemonics (vertical and horizonal)

1. We are still in GNDRW and the current picture is of the resistor but we could call it off the disk by typing "##name."
2. *L (in light button area)--destroy local lines. This will eliminate the old line image of the resistor.
3. *C--delete comment text.
4. *B--bring box onto screen.
5. *T--move terminals. Hit a point on the top of the box and then POINT A. Do the same for the bottom of the box and POINT B. This will move both terminals.
6. *PANIC--the picture will be regenerated with the terminals moved.
7. *REORG, *either terminal--in line drawing mode, connect the two terminals by drawing a vertical resistor.
8. *REORG, *a point on the side of the resistor indicating the parameter model list location.
9. Type C#--create mnemonic--now in the text editor. Enter from the keyboard a list of parameters (i.e., the resistance, R in this case).

10. *RETURN--back in GNDRW.
11. Type "S&name.0"--this stores the mnemonic.
12. Now construct a horizontal mnemonic. Type "##name" to bring the original picture back (or start at step B2).
13. *C--destroy comment text.
14. *REORG, POINT C--set an origin for the location of parameter list.
15. Type "C#"--create mnemonic. Now in the text editor. Type the parameter "R" followed by the RETURN key.
16. *RETURN--to GNDRW.
17. Type "S&name.1"--this will save the horizontal mnemonic. Notice that the picture could have been redrawn and the terminals moved as in the vertical resistor case, if a larger or smaller mnemonic was desired.

C. Create a resistor for analysis (GSAMV1)

1. *RETURN--get back to the top level of the system and select the simulation and modeling system, *GSAMV1.
2. *CONSTR--now in GNDRW again but this time within the modeling system. The entire section A could be repeated, but an easier way is to type "##name" which will bring the completed picture off of the disk.
3. *RETURN--back to the top level of GSAMV1. The drawing is complete so now the analysis information must be specified by: *SPECFY. In the specification program, the different data types appear in the menu area.
4. *DEFS--define terminal types. Type "ELECTRIC:E(VOLT), I(CURRENT)" followed by the RETURN key.
5. *REGEN--the definition appears in the upper left of the screen.
6. *RETURN--back to SPECFY to choose another variable.
7. *EQATNS--enter all equations.

8. Type "VOLT(O)=VOLT(1) + CURRENT(1) * R" and "CURRENT(1)=-CURRENT(O)" both followed by the RETURN key. Note that the variables defined in DEFS refer to the two terminals (0,1) through subscripting.
9. *RETURN--back to SPECFY.
10. *PARMS--enter parameters.
11. Type "R=100" followed by the RETURN key.
12. *REGEN--"R=100" appears in the upper left of the screen.
13. *RETURN--back to SPECFY. No more data is desired for this particular example.
14. *RETURN--back to GSAMV1.
15. *CONSTR--into GNDRW.
16. Type "S#name" which will save the picture and the data specification.
17. Now the picture and two mnemonics are done and have been stored on the disk. The program is terminated by: *RETURN (to GSAMV1), *RETURN (to GRASS), *LØGØUT.

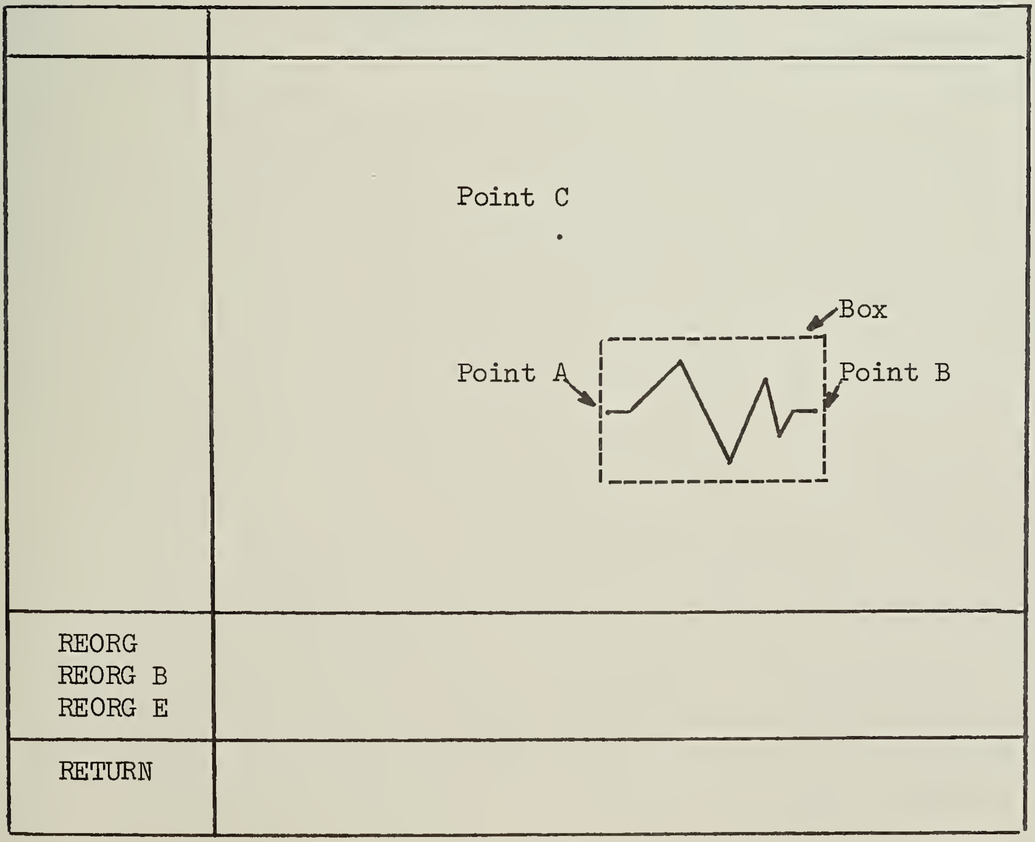


Figure 3. Resistor Example

APPENDIX B

Loading the System

One part of the system must be loaded manually into the PDP-8 from two magnetic tape units located to the right of that computer's control panel. Another part of the system is loaded automatically into the PDP-8/I from the PDP-8; however, a small bootstrap loader must be present in the PDP-8/I. This code must be loaded manually into the PDP-8/I from a paper tape unit located above that computer's control panel. Figure 4 shows the control panel switches for both computers, while Figure 5 describes one of the magnetic tape units. A teletype operator's console is attached to each computer. Short discussions of entering and examining data in the core memory of the computers and of readying a magnetic tape unit are followed by a detailed outline for loading the system.

Consult Figure 4. To enter or examine data in the memory of either computer, the address of the desired data must first be set. Press the STØP switch, then set the switch-register switches to the desired number, for example "7756." Press LØAD ADD (load address)--the address is now ready. To enter data, set the switch-register switches to the proper value, for example, "6014," and press DEP (deposit). The value "6014" has now been entered at location "7756" and the address has been automatically incremented by one. Set the switch-register switches for the next value and press DEP again. Continue in this manner until all desired data is entered. To examine data, set the address as above and press EXAM (examine). The contents of the memory location will appear in the row of lights labeled "memory buffer." The address is automatically incremented by one every time the EXAM key is pressed. Hence, to view the next sequential location, simply press EXAM again.

Consult Figure 5. To remove a tape from the unit, first check that all the tape is on the left reel. If this is NOT the case, put the TAPE CONTROL switch to LØCAL and hold the ← (reverse) switch down until all the tape is on the left reel. Put the TAPE CONTROL switch to ØFF (neither the top nor the bottom of the switch is depressed). Now pry the left reel off the hub. To mount a tape on an empty unit, first check that the WRITE LØCK switch is on (e.g. the tape cannot be overwritten). Press the reel

firmly down over the left hub until secure; a light "snap" will be heard and the words "Digital Equipment Corporation" will be visible on the face of the reel. Thread the tape over the left tape guide, over the read/write head, over the right tape guide, and wrap the end around the righthand reel about five times. Set the TAPE CONTROL switch to LØCAL and hold the → (forward) switch down for several seconds, thus, winding leader onto the takeup (righthand) reel. Now set the TAPE CONTROL switch to REMØTE. If subsequent use during setup causes the tape to rewind entirely onto the left reel, inadequate leader was used, and the entire procedure must be restarted. If a rewound reel is spinning, make sure that the TAPE CONTROL switch is set to ØFF, then the reel can be stopped with light hand pressure on the rim.

Entering the loader into the PDP-8/I is done as follows

1. Press the STØP switch.
2. Check that the "Rim Loader" is in core at location 7756. Using EXAM as previously described, compare the memory contents with the "Rim Loader" slip mounted on the control panel. If the memory contents are not correct, use DEP as previously described to enter the "Rim Loader" as defined on the slip.
3. Set the switch-register switches to "7756" and press LØAD ADD.
4. Raise the paper tape reader tape-guide and slip the tape marked "Binary Loader" under the read head from the righthand side and onto the sprocket wheel. Lower the tape-guide, making sure that the sprocket holes in the tape properly mesh with the sprockets on the wheel. Press the FEED switch for just an instant to ensure that the tape is moving properly. Set the READ switch to ØN. Press the START switch; the tape will feed through to the left side of the reader.
5. When tape motion has ceased, press the STØP switch. Remove the tape by using the FEED switch to flush any trailer. Set the switch-register switches to "7777" and press LØAD ADD. Now set the switch-register switches to "3777" (e.g. just turn off the "high" bit). DO NOT press LØAD ADD.

6. Mount the tape marked "MØN8/I" as described above for the "Binary Loader"; then press the START switch.
7. When tape motion has ceased, press the STØP switch and remove the tape.
8. Set the switch-register switches to 7600 and the instruction field switches to 01 (e.g., the address is 017600). Note that for all previous operations, the data field and instruction field switches should have been set to zero. Press LØAD ADD; then press START. The bootstrap loader is now executing in the PDP-8/I. Turn the switch on the PDP-8/I's teletype to LINE.

Loading the system into the PDP-8 is done as follows:

1. Press the STØP switch. Mount the GRASS library tape on the top magnetic tape unit and the GRASS load tape on the bottom unit. Rotate the top unit's unit-select cylinder until a "4" appears in the center; set the bottom unit's cylinder to "8."
2. Set the switch-register switches to "7577" and press LØAD ADD. Then set the switch-register switches to "0000." DO NOT press LØAD ADD.
3. Raise the IPL switch. After the lights on the control panel have stopped flickering, lower the IPL switch.
4. Set "7600" into the switch-register switches and press LØAD ADD.
5. Turn the switch on the PDP-8's teletype to LINE. Set the WRITE ENABLE switch on the unit-8 tape unit on (e.g., the tape can be overwritten). Press the START switch. The unit-8 tape will move and then stop; this will be followed by a carriage-return/line-feed on the teletype.
6. Type "GLØAD" followed by carriage-return on the teletype. After the unit-8 tape begins to wind forward, set the WRITE LØCK switch on. The first part of the system is now being loaded onto the disk.* (see starred information page (36).

7. The question "RESTØRE LIBRARY?" will eventually appear on the teletype. Valid responses are:

   "T": restore disk library from unit-4 tape.
   "D": use current library on the disk.
   "N": initialize disk library to "empty."

   Note that for responses "D" and "N", the GRASS library tape need not have been mounted on unit-4. If "T" is the response, the copy of the library on the unit-4 tape will now be transferred to the disk.* (see starred information below).

8. The question "DRØP ACID?" will eventually appear on the teletype. Valid responses are:

   "Y": automatically transfer code to the PDP-8/I.
   "N": omit transfer since the PDP-8/I is already loaded.

   Note that "Y" is the proper response when the PDP-8/I has be re-initialized as described previously. "N" is the usual response for restarting the PDP-8 and when the PDP-8/I has remained undisturbed.

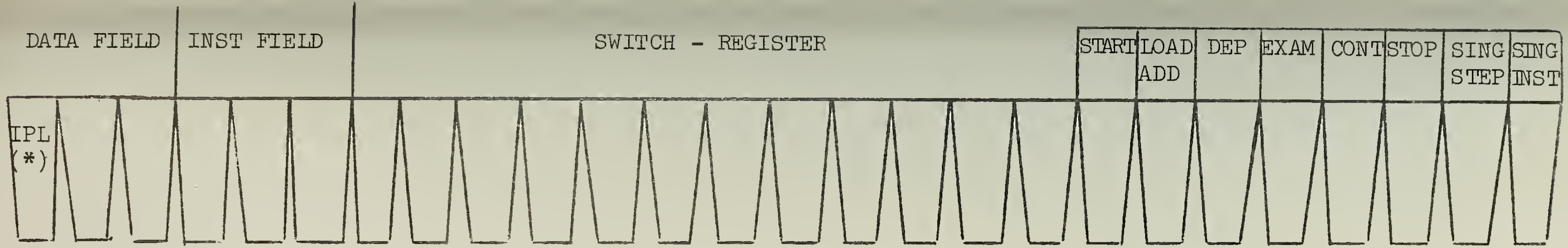
9. The message "WELCØME TØ THE WØNDERFUL WØRLD ØF GRASS" will appear on the PDP-8 teletype. In addition, if "Y" was the response in step 8, the message "WELCØME TØ THE WØNDERFUL WØRLD ØF ACID" will appear on the PDP-8/I teletype. When both tapes have stopped moving, set both of the TAPE CONTROL switches to ØFF.

10. The system is ready.

* While data is being transferred from tape to disk (steps marked by asterisks above and below), error checking is being performed. Detection of a data error is indicated by a bell ring on the PDP-8 teletype. If a continuous ringing occurs, press STØP and have the operator examine the system in its current state. An occassional ring, however, is all right.

At the end of a session, when no terminals are in used the library can be saved on the unit-4 tape as follows:

1. Press STØP--then perform steps 4 and 5 of the above PDP-8 load sequence.
2. Set the WRITE ENABLE switch on unit-4 on. Type "GEXIT" followed by carriage-return on the teletype. The library will now be written onto the unit-4 tape.*
3. The unit-4 tape will be rewound and a carriage-return/line-feed will be printed on the teletype. When rewinding begins, the WRITE LØCK switch on unit-4 should be set on. Dumping the library is now complete.
4. Dismount all tapes. Turn off all teletypes and terminals. Note that the system could be restarted with the current library after step 3. Simply type "GLØAD" carriage-return, and reply "D" and then "N."



* <u>PDP-8 ONLY</u>

Note: The data field, inst field, and switch-register switches have two positions. The "rocker" type switches on the PDP-8/I indicate a "one" when the bottom of the switch is pressed down. The "lever" type switches on the PDP-8 indicate a "one" when raised and a "zero" when depressed. All other switches are single positional; that is, they return to their original position after being pressed. The orientation of the DEPØSIT key is purposely inverted to emphasize its use.

Figure 4. PDP-8 and 8/I Control Panel Switches

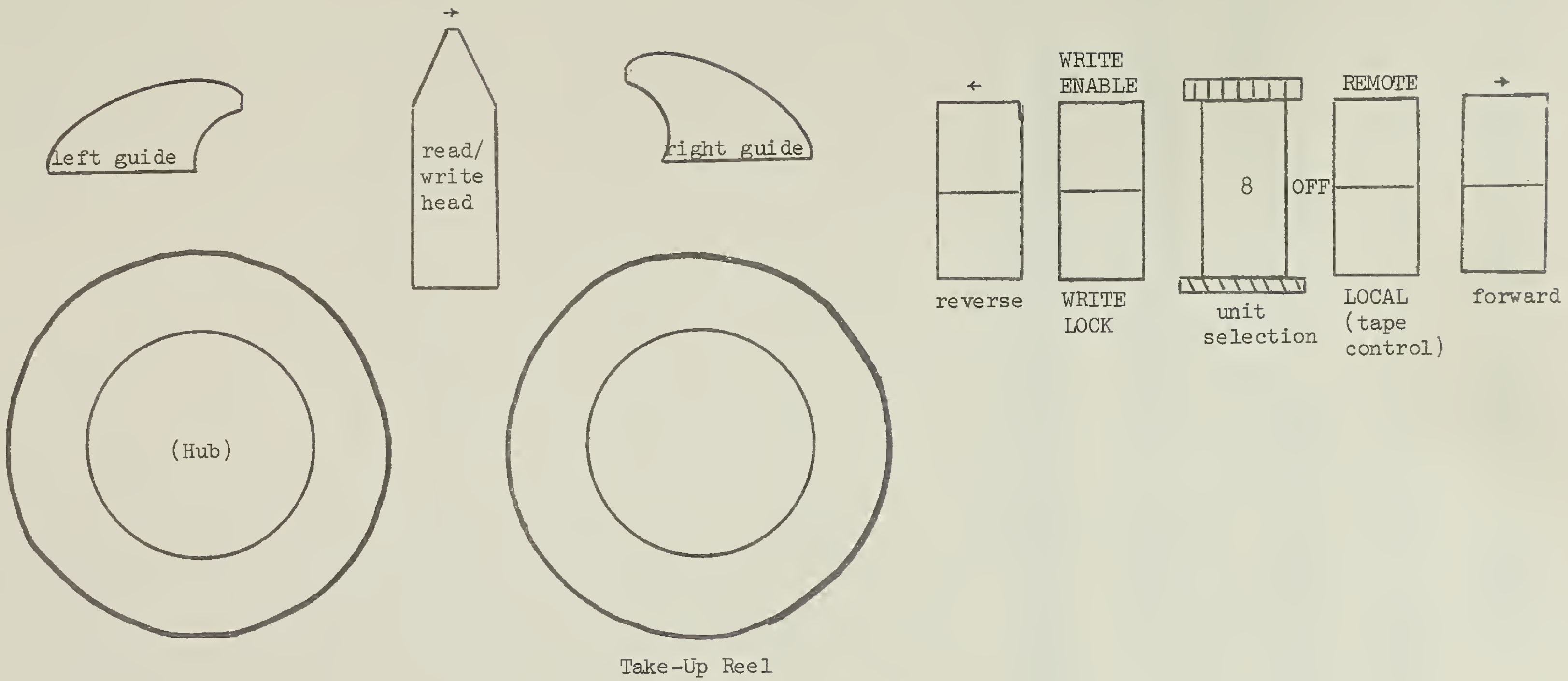


Figure 5. DECtape Unit Detail

LIST OF REFERENCES

[1] Michel, M. J. GRASS: System Overview, Department of Computer Science Report No. 465, University of Illinois, Urbana, Illinois 61801, July 1971.

[2] Computek User's Manual, Series 400/15 CRT Display System, Computek Inc., Bulletin 400M, Chapter 3, pp. 3-1 through 3-10, July 1969.

[3] Haskin, R., Nickolls, J., and Michel, M. J. GRASS: Remote Facilities Guide, Department of Computer Science Report No. 466, University of Illinois, Urbana, Illinois 61801, July 1971.

[4] Michel, M. J. GRASS: System Software Description, Department of Computer Science Report No. 468, University of Illinois, Urbana, Illinois 61801, August 1971.

[5] Gear, C. W., et.al. THE SIMULATION AND MODELING SYSTEM--A Snapshot View, Department of Computer Science File No. 824, University of Illinois, Urbana, Illinois 61801, February 1970.

m AEC–427
(6/68)
ECM 3201

U. S. ATOMIC ENERGY COMMISSION
UNIVERSITY–TYPE CONTFACTOR'S RECOMMENDATION FOR DISPOSITION OF SCIENTIF:C AND TECHNICAL DOCUMENT

(See Instructions on Reverse Side)

AEC REPORT NO.
COO-1469-0189

2. TITLE
GRASS: Terminal User's Guide

TYPE OF DOCUMENT (Check one):

☒ a. Scientific and technical report
□ b. Conference paper not to be published in a journal:
Title of conference ______
Date of conference ______
Exact location of conference ______
Sponsoring organization ______
□ c. Other (Specify) ______

RECOMMENDED ANNOUNCEMENT AND DISTRIBUTION (Check one):

☒ a. AEC's normal announcement and distribution procedures may be followed.
□ b. Make available only within AEC and to AEC contractors and other U.S. Government agencies and their contractors.
□ c. Make no announcement or distrubution.

REASON FOR RECOMMENDED RESTRICTIONS:

SUBMITTED BY: NAME AND POSITION (Please print or type)
C. W. Gear, Professor
and Principal Investigator

Organization
Department of Computer Science
University of Illinois
Urbana, Illinois 61801

Signature Charles W Gear.

Date August 1971

FOR AEC USE ONLY

AEC CONTRACT ADMINISTRATOR'S COMMENTS, IF ANY, ON ABOVE ANNOUNCEMENT AND DISTRIBUTION RECOMMENDATION:

PATENT CLEARANCE:

□ a. AEC patent clearance has been granted by responsible AEC patent group.
□ b. Report has been sent to responsible AEC patent group for clearance.
□ c. Patent clearance not required.